

In the claims

1. (Original) A computer-implemented method for persisting object-oriented data objects, comprising:

- defining in a source program a first structure type for storage of one or more data values;
- defining in the source program a plurality of objects of the structure type;
- initializing the objects with selected data values in the source program;
- defining in the source program a first class that derives from the structure type, the class including a static method configured to convert an object of the structure type to an instance of the class in response to a reference to the method; and
- allocating memory for the objects and assigning the data values to the objects at compile-time.

2. (Original) The method of claim 1, further comprising:

- defining in the source program one or more traits classes, each traits class including first, second, and third public type definitions, the first type definition identifying a class described by the traits class, the second type definition identifying a base structure for the class described by the traits class, and the third type definition for identification of a structure for input to the static method of the class described by the traits class, for reference to the class described by the traits class, and for assignment of data to the class described by the traits class; and
- defining in the source program a second class, the second class including an attribute of the third type definition of a traits class that describes the first class.

3. (Original) The method of claim 2, further comprising:

- defining in the source program a collection class that includes objects of the first class;
- and
- instantiating objects of the first class in an object of the collection class in the source program.

4. (Original) The method of claim 3, further comprising defining in the source program a collection iterator configured to traverse the objects of the first class in the collection object.
5. (Original) The method of claim 4, further comprising:
defining in the source program a second structure type that includes one or more void pointers and an input aggregate structure; and
defining one or more of the traits classes with the second structure type used as the third public type definition.
6. (Original) The method of claim 5, further comprising:
defining a template class that includes an initialize method that calls a language-provided function that constructs an object of a specified type;
defining in one of the traits classes a member attribute that is a pointer to the initialize method.
7. (Original) The method of claim 4, further comprising:
defining in the source program a pre-processor macro that encapsulates the collection class; and
instantiating objects of the first class in an object of the collection class in the source program by reference to the pre-processor macro.
8. (Original) The method of claim 7, further comprising defining in the source program a pre-processor macro that forward references the collection class.
9. (Original) The method of claim 7, further comprising defining in the source program a pre-processor macro that references an element of a first collection in an entry of a second collection.
10. (Original) The method of claim 7, further comprising defining in the source program a pre-processor macro that initializes a collection having no entries.

11. (Original) The method of claim 7, further comprising:

defining in the source program a second structure type that includes one or more void pointers and an input aggregate structure; and

defining one or more of the traits classes with the second structure type used as the third public type definition.

12. (Original) The method of claim 11, further comprising:

defining a template class that includes an initialize method that calls a language-provided function that constructs an object of a specified type;

defining in one of the traits classes a member attribute that is a pointer to the initialize method.

13. (Original) The method of claim 4, further comprising defining an initializer class that iterates through objects in the collection class invoking a default constructor for an input class type.

14. (Original) The method of claim 2, further comprising:

defining in the source program a second structure type that includes one or more void pointers and an input aggregate structure; and

defining one or more of the traits classes with the second structure type used as the third public type definition.

15. (Original) The method of claim 10, further comprising:

defining a template class that includes an initialize method that calls a language-provided function that constructs an object of a specified type;

defining in one of the traits classes a member attribute that is a pointer to the initialize method.

16. (Original) An apparatus for persisting object-oriented data objects, comprising:

means for defining in a source program a first structure type for storage of one or more data values;

means for defining in the source program a plurality of objects of the structure type;
means for initializing the objects with selected data values in the source program;
means for defining in the source program a first class that derives from the structure type,
the class including a static method configured to convert an object of the structure type to an
instance of the class in response to a reference to the method; and
means for allocating memory for the objects and assigning the data values to the objects
at compile-time.